Install

from archlinux iso

- script installer

```
systemctl enable dhcpd
```

install on archiso zfs system from https://github.com/eoli3n/archiso-zfs

```
curl -s https://raw.githubusercontent.com/eoli3n/archiso-zfs/master/init |
bash
```

ZFS

Reset disk and create GPT partition schema

```
DISK=/dev/disk/by-id/ata-HFS512G39MND-3510A_FJ64N5235113A4S13
sgdisk --clear -g $DISK
```

Prepare partitions

```
# Run this if you need legacy (BIOS) booting:
# sgdisk -a1 -n2:34:2047  -t2:EF02 $DISK

# Run this for legacy (BIOS) boot or UEFI booting or raidz pool:
sgdisk     -n3:1M:+512M -t3:EF00 $DISK
parted $DISK set 3 bios_grub on # needed for embedding grub in this
partition

# and after run these:
sgdisk     -n9:-8M:0    -t9:BF07 $DISK
sgdisk     -n1:0:0      -t1:BF01 $DISK
```

Create pool or ...

```
zpool create -o ashift=12 \
      -O atime=off -O canmount=off -O normalization=formD \
      -O mountpoint=/ -R /mnt -f \
      rpool ${DISK}-part1
```

... import existing pool

```
zpool export rpool
zpool import -R /mnt rpool
... umount everything inside /mnt and remove subfolders
```

```
zfs mount rpool/ROOT/arch
```

or create raidz pool

```
zpool create -O atime=off -O mountpoint=/ -R /mnt -O canmount=off -O
compression=lz4 -O normalization=formD -o ashift=12 \
  rpool raidz /dev/disk/by-id/ata-ST4000NM0035-1V4107_ZC10????-part1
```

create dataset

```
zfs create -o canmount=off -o mountpoint=none rpool/ROOT
zfs create -o canmount=noauto -o mountpoint=/ rpool/ROOT/arch
zfs mount rpool/ROOT/arch
zfs create -o setuid=off rpool/home

zpool set bootfs=rpool/ROOT/arch rpool
```

install base system

```
pacstrap /mnt base
echo "rpool/ROOT/arch      /             zfs
rw,noatime,xattr,noacl    0 0" > /mnt/etc/fstab
```

zfs cache

```
[ -f /etc/zfs/zpool.cache ] || zpool set cachefile=/etc/zfs/zpool.cache
rpool
mkdir -p /mnt/etc/zfs
cp /etc/zfs/zpool.cache /mnt/etc/zfs/zpool.cache
```

```
arch-chroot /mnt
```

from existing linux

- https://wiki.archlinux.org/index.php/Install_from_existing_Linux#Method_B:_Using_the_LiveCD_image
- https://golem.linux.it/wiki/Installare_Arch_Linux_su_ZFS
- https://ramsdenj.com/2016/06/23/arch-linux-on-zfs-part-2-installation.html (great !!!!)

Install system

```
pacman -Syyu
pacman -S base vim
```

add archzfs repo

```
cat >> etc/pacman.conf <<EOF
```

```
[archzfs]
Server = http://archzfs.com/\$repo/x86_64
EOF

pacman-key --init
pacman-key -r F75D9D76
pacman-key --lsign-key F75D9D76
pacman -Syyu
pacman --noconfirm -S linux-lts linux-headers
pacman --noconfirm -S zfs-dkms
systemctl enable zfs.target
systemctl enable zfs-import-cache
systemctl enable zfs-mount
systemctl enable zfs-import.target
```

basic configuration

```
ln -sf /usr/share/zoneinfo/Europe/Rome /etc/localtime
hwclock --systohc
cat > /etc/locale.gen <<EOF
en_US.UTF-8 UTF-8
it_IT.UTF-8 UTF-8
EOF
locale-gen

echo LANG=it_IT.UTF-8 > /etc/locale.conf

echo nero > /etc/hostname

echo "KEYMAP=it\nFONT=Lat2-Terminus16" > /etc/vconsole.conf
```

edit /etc/mkinitcpio.conf to change HOOKS order

```
HOOKS=(base udev autodetect modconf block keyboard zfs filesystems)
```

regenerate initram

```
mkinitcpio -p linux-lts
```

bios grub

```
pacman -S grub os-prober
ZPOOL_VDEV_NAME_PATH=1 grub-install --target=i386-pc /dev/sda
ZPOOL_VDEV_NAME_PATH=1 grub-mkconfig -o /boot/grub/grub.cfg
```

uefi grub

```
# exit from chroot
mount /dev/sda3 /arch/efi # /dev/sda3 is EFI partition
chroot /arch bash
```

```
mount -t efivarfs efivarfs /sys/firmware/efi/efivars
pacman --noconfirm -S  grub efibootmgr
ZPOOL_VDEV_NAME_PATH=1 grub-install --target=x86_64-efi --efi-directory=/efi
--bootloader-id=arch
```

edit /etc/default/grub

```
GRUB_CMDLINE_LINUX_DEFAULT="noresume zfs_force=1 libahci.ignore_sss=1
vga=current"
```

generate grub config

```
export ZPOOL_VDEV_NAME_PATH=1
grub-mkconfig -o /boot/grub/grub.cfg
```

root password

```
passwd
```

User

```
pacman -S sudo
useradd -m scipio
usermod -a -G wheel scipio
passwd scipio
```

decomment wheel in /etc/sudoers

Basic

```
systemctl start systemd-timesyncd.service
systemctl enable systemd-timesyncd.service
```

Gnome

```
pacman -S gnome gpaste network-manager-applet
systemctl enable NetworkManager.service
systemctl enable gdm
```

nvidia

```
pacman -S nvidia-dmks
```

others

```
pacman -S fish rsync openssh
```

```
pacman -S libreoffice-fresh-it
```

```
exit
mount | grep -v zfs | tac | awk '/\/mnt/ {print $3}' | xargs -i{} umount -lf
{}
zpool export rpool
reboot
```